

Analisis Kualitas Layanan Jaringan Seluler 5g Privat Berbasis *Cloud Computing*

1st Repka Mochamad Zaqy Mulyadi

Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

repkazaqy@student.telkomuniversity.ac.id

2nd Rendi Munadi

Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

rendymunadi@telkomuniversity.ac.id

3rd Fardan

Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia

fardanfnn@telkomuniversity.ac.id

Abstrak— Perkembangan dunia telekomunikasi selalu berlangsung dengan cepat. Salah satu perkembangan tersebut adalah teknologi seluler generasi kelima atau 5G. Salah satu implementasi adalah penggunaan jaringan 5G privat yang dilakukan dalam bentuk sebuah proyek *open-source* virtualisasi 5G core network. 5G core network yang digunakan bernama Free5GC dan OpenAirInterface Keduanya menjalankan core dan access network dalam platform Cloud Computing yang deploymentnya dibantu dengan kontainerisasi Docker. Dari hasil pengujian Quality of Services yang dilakukan penulis sesuai dengan keempat parameter yang diukur yaitu Throughput, Delay, Jitter, dan Packet Loss. Penulis melakukan tes dengan tiga skema pengguna yaitu 1 UE, 3 UE, dan 5 UE yang dilakukan dalam waktu 15 detik. Hasil pengukuran Quality of Services dari kedua jaringan telah dilakukan dan pada jaringan OpenAirInterface dengan skenario 1,3, dan 5 UE didapat Throughput sebesar 1.5566 Gbps, 1.5103 Gbps, dan 1.4283 Gbps. Sedangkan Free5GC dengan skenario 1,3, dan 5 UE didapat 1.9782 Gbps, 1.8721 Gbps, dan 1.6591 Gbps. Untuk Delay OpenAirInterface didapat 0.1537 ms, 0.2002 ms, dan 0.2172 ms. Pada Free5GC didapat 0.1114 ms, 0.1491 ms, dan 0.1764 ms. Hasil untuk Jitter pada OpenAirInterface didapat 0.15379 ms, 0.20022 ms, dan 0.21728 ms. Sedangkan Free5GC didapat 0.11145 ms, 0.14911 ms, dan 0.17643 ms. Untuk Packet Loss di kedua jaringan didapat hasil 0% atau tidak ada.

Kata kunci— jaringan 5G cloud, kontainerisasi, quality of services, free5GC, openairinterface.

I. PENDAHULUAN

Pertukaran informasi secara cepat merupakan dambaan setiap orang di era modern ini. Hal tersebut semakin terwujud dengan dimulainya era generasi kelima dalam perkembangan jaringan seluler atau yang disebut dengan teknologi 5G. Berbagai pengembangan terus dilakukan guna mengoptimalkan teknologi tersebut dari berbagai aspek, dimulai dari sisi layanan, infrastruktur, regulasi, sampai berbagai inovasi yang dapat dimasukkan untuk membuat layanan 5G semakin efektif membantu banyak orang menyampaikan sesuatu lebih cepat dibanding teknologi sebelumnya. Dalam proses pemutakhiran tersebut perlu banyak penelitian dan percobaan terutama simulasi, berbagai lembaga mulai mengembangkan simulasi jaringan 5G baik yang bersifat *open source* maupun komersial. Dalam penelitian ini dilakukan simulasi jaringan seluler 5G menggunakan jaringan 5G di sisi core network dan juga access network yang melibatkan Free5GC[1] dan OpenAirInterface[2] yang merupakan *open-source* 5G core yang dikembangkan untuk di sisi core, lalu untuk di sisi

access digunakan UERANSIM atau simulator untuk UE (user end) dan RAN (radio access network). Ketiga proyek tersebut merupakan proyek yang bersifat *open source* dan dibuat untuk keperluan pengembangan jaringan 5G kedepannya.

Jaringan 5G memiliki beberapa tujuan yang berbeda dengan generasi jaringan sebelumnya, dua aspek yang paling disorot yaitu terkait penggunaan daya[3] yang dibuat untuk menjadi lebih hemat serta melakukan pembangunan komponen atau infrastruktur seminim mungkin agar menghemat biaya pembangunannya.[4] Kedua hal tersebut tidak boleh membuat jaringan 5G menjadi berkurang kualitasnya, namun harus tetap terjaga atau bahkan lebih baik lagi. Dalam pengembangannya, ada beberapa kolaborasi yang dilakukan terhadap jaringan 5G ini, salah satu bentuknya yaitu memvirtualisasikan fungsi jaringan atau biasa disebut dengan NFV atau kepanjangan dari Network Function Virtualization.. Ketika bicara NFV maka kita membutuhkan sebuah wadah yang bisa menjadi induk bagi core network jaringan 5G ini, dalam penelitian ini layanan Cloud Computing dipilih sebagai wadah tersebut karena fleksibilitas dan visi dari layanan ini bisa digunakan sebagai tempat bagi komponen core network untuk disimulasikan.[1] Dalam penelitian ini, bagian dari core network dan user di-deploy atau disebar ke layanan Cloud Computing dengan menggunakan metode kontainerisasi dan virtual machine, layanan kontainer yang digunakan yaitu Docker.[5] Dengan melakukan deploy bagian core dari jaringan dengan kedua cara tersebut, tentunya akan membantu efisiensi layanan sesuai dengan tujuan dikembangkannya jaringan generasi kelima ini.[4]

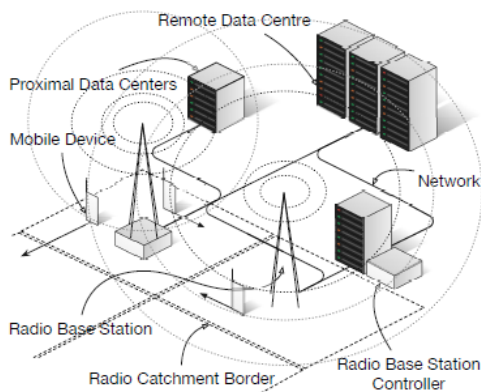
Kedua metode yang disebutkan sebelumnya merupakan tata cara untuk menunjang penggunaan jaringan seluler 5G secara privat. Walau menggunakan kedua cara tersebut yang membuat bagian core menjadi virtual, maka harus tetap diperhatikan aspek kualitas layanan atau Quality of Services yang sering disingkat dengan QoS. Hal tersebut sudah menjadi ketetapan guna mencapai kenyamanan saat menggunakan layanan sekalipun layanan yang digunakan bersifat privat. Maka dari itu, hal tersebut mendorong penulis untuk melakukan analisis terhadap penggunaan konsep tersebut dari sisi Quality of Services jaringan tersebut[6]. Penelitian ini dilakukan dengan membandingkan Quality of Services setelah melakukan deploy terhadap jaringan 5G di layanan Cloud Computing melalui metode deploy Docker dan

Virtual Machine. Setelah melakukan analisis tersebut penulis menargetkan keluaran dari penelitian ini bisa menjadi referensi bagi siapa pun yang ingin mengimplementasikan jaringan 5G secara privat maupun membantu melakukan efisiensi terhadap infrastruktur suatu jaringan 5G ini, mengingat masih kurangnya referensi tentang penelitian ini.

II. KAJIAN TEORI

A. Telco Cloud

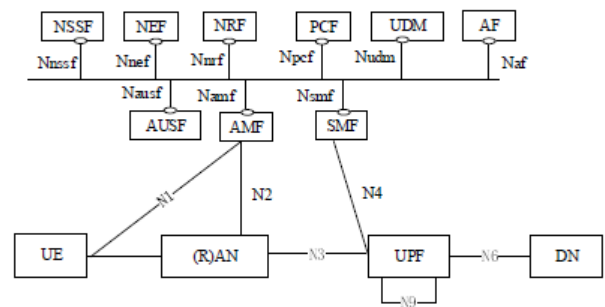
Jaringan telekomunikasi berbasis *Cloud* merupakan sebuah inovasi dengan banyak manfaat. Manfaat utama yang paling jelas adalah dampak terhadap biaya (*cost*) dan kualitas layanan (QoS).[7] Dengan menggunakan konsep *Telco Cloud* penyedia jasa telekomunikasi dapat melakukan virtualisasi terhadap perangkat di bagian *core* dari jaringan yang mereka miliki, hal tersebut tentunya mengubah perangkat yang awalnya berbentuk fisik nyata menjadi virtual atau berbasis perangkat lunak. Setelah itu perangkat tersebut disebar (*deploy*) ke layanan *cloud* guna memperluas cakupan penggunaan jaringan, dengan begitu biaya yang perlu dikeluarkan lebih murah karena hanya harus menyewa server ke *platform cloud* bukan membeli sebuah perangkat. Namun permasalahan dalam konsep *Telco Cloud* sendiri terjadi pada tidak stabil di ranah *Quality of Services* (QoS) yaitu terjadinya ketidakstabilan *latency*. Skema *Telco Cloud* bisa dilihat pada Gambar 2.1. [7]



GAMBAR 2.1 Skema *Telco Cloud* [7]

B. Jaringan Seluler 5G Privat

Perkembangan jaringan telepon seluler telah sampai di generasi kelima dan kita biasa menyebutnya dengan “5G” atau singkatan dari *fifth-generation*. Jaringan seluler ini merupakan jaringan terbaru yang memiliki banyak tujuan dalam implementasinya. 5G dikembangkan untuk membantu menopang infrastruktur digital seperti *Internet of Things* yang akan digunakan oleh massa yang lebih banyak, kemudian otomatisasi sistem seperti contohnya di bidang kedokteran dan industri, selain itu 5G pun bertujuan untuk membuat jaringan ini menjadi lebih murah atau *cost* rendah dan juga ramah lingkungan dalam penggunaannya.[3]



GAMBAR 2.2 Arsitektur Jaringan 5G

Setiap komponen jaringan 5G di atas memiliki peranan yang penting terutama 3 komponen utama yaitu UPF, SMF, dan AMF yang menjadi ‘jangkar’ atau penghubung *access* dan *core network* jaringan 5G. Berikut ini merupakan ringkasan fungsi dari setiap komponen jaringan 5G yang tertera pada Gambar 2.2 di atas:

1. **AUSF:** *Authentication Server Function* atau AUSF berfungsi untuk menyimpan ‘key’ atau kunci yang nantinya akan digunakan kembali untuk otentikasi dimana sebelumnya kunci tersebut didapat dari pendaftaran UE pada jaringan ini[10]. AUSF juga menangani permintaan otentikasi sesuai standar 3GPP *access* maupun *untrusted 3GPP access*.
2. **AMF:** *Access and Mobility Management Function* atau AMF memiliki fungsi yang secara umum yaitu mengontrol sinyal antara *core network* dengan perangkat pengguna. Selain itu AMF juga memiliki beberapa fungsi lain seperti menjaga keamanan jaringan dan data pengguna.[10]
3. **AF:** *Application Function* atau AF memiliki fungsi sebagai bagian yang menyediakan layanan aplikasi bagi pengguna atau *user*. Sebagai contohnya layanan ini bisa memberikan sejenis layanan *streaming video* kepada pengguna[10].
4. **SMF:** *Session Management Function* atau SMF memiliki fungsi yaitu untuk melakukan manajemen terhadap alokasi alamat IP kepada *user equipment* atau UE. Selain itu fungsi lainnya adalah sebagai autentikator untuk otentikasi dan otorisasi disaat ada permintaan UE untuk melakukan sesi PDU pada jaringan 5G[10].
5. **NEF:** *Network Exposure Function* atau NEF memiliki fungsi sebagai bagian yang mengekspos informasi kemampuan fungsi dari *core network* ke eksternal jaringan. Bentuk dari ekspos informasi tersebut bisa dilihat karena bisa dilakukan untuk memantau konektivitas UE, pergerakan UE, dan membantu analisis QoS jaringan tersebut[10].
6. **NRF:** *Network Repository Function* atau NRF memiliki fungsi sebagai bagian yang digunakan untuk mengeksplorasi fungsi jaringan yang ada dalam jaringan tersebut. Selain itu bagian ini juga bisa digunakan untuk mengeksplorasi fungsi jaringan baru juga apabila ditemukan[10].
7. **PCF:** *Policy Control Function* atau PCF merupakan sebuah bagian dalam jaringan 5G yang memiliki fungsi sebagai pengawas dan pembuat peraturan dalam lingkup bagian atau komponen *Control Plane*, seperti AMF contohnya[10].
8. **UDM:** *Unified Data Management* atau UDM memiliki

fungsi sebagai penghasil otoritas *Authentication and Key Agreement* atau AKA. Otoritas tersebut memberikan akses untuk jaringan dalam mengakses data pelanggan, langganan, manajemen SMS, dan manajemen penyimpanan SUPI [10].

9. **UPF:** *User Plane Function* atau UPF, merupakan bagian yang memiliki fungsi sebagai gerbang untuk gNB atau RAN dari jaringan 5G ke dunia luar, kemudian ia memiliki fungsi sebagai bagian yang melakukan *routing* atau perutean terhadap paket data, penyaringan paket data, inspeksi paket data, dan titik jangkar untuk mobilitas antar RAT [10].
10. **RAN:** *Radio Access Network* atau biasa disingkat RAN yang memiliki fungsi sebagai penghubung antara *core* dengan *access network*. RAN biasanya terhubung langsung dengan UE, UPF, dan AMF. Dalam jaringan 5G, gNB adalah RAN dalam jaringan ini yang nantinya terhubung dengan salah satunya yaitu UE[10].
11. **UE:** UE atau *User Equipment* merupakan perangkat yang digunakan oleh pelanggan atau *user*. Contoh dalam penggunaan sehari-hari yaitu seperti telepon genggam. Makna UE menggambarkan 'perangkat yang digunakan' oleh pengguna dan UE biasanya langsung terhubung ke gNB (RAN) dan juga AMF[10].

Semua perangkat di atas sudah termasuk dalam ranah *core network* maupun *access network*, komponen ini hanya dapat ditemukan pada jaringan 5G dengan jenis *standalone*.

C. Cloud-Native Network Function

Migrasi dari konsep *Network Function Virtualization* (NFV) ke layanan *cloud* disebut dengan *Cloud-Native Network Function* atau CNF.[1] Dari beberapa jenis layanan *cloud*, migrasi NFV ke layanan *cloud* digolongkan sebagai jenis *Infrastructure as a Service* atau IaaS. Alasannya karena konsep IaaS menyediakan layanan *cloud* yang dapat mengelola perangkat keras yang sudah divirtualisasikan dan dimigrasikan ke layanan *cloud*[11]. Sebagai contohnya, metode kontainerisasi digunakan untuk migrasi ke layanan *cloud*.

Dalam *Cloud-Native* kita diharuskan memilih penyedia layanan *cloud* sesuai dengan kebutuhan kita, penyedia layanan *cloud* disebut dengan *Cloud Platform*. Dari sekian banyak *Cloud Platform* kebanyakan orang memilih berdasarkan penyedia yang dapat membuat mereka nyaman, padahal untuk memilih penyedia terutama untuk CNF diperlukan *Cloud Platform* yang cocok dengan layanan yang akan kita pakai untuk *software* atau infrastruktur kita[12]. Maka dari itu dalam penelitian ini, penulis menggunakan layanan *Azure* milik *Microsoft* sebagai penyedia layanan *cloud* yang dipilih. Alasannya layanan *cloud* ini terbilang presisi terutama dari sisi ketersediaan server, penyimpanan, konfigurasi manual dan biaya yang dibebankan.[13]

Metode kontainerisasi merupakan konsep *deployment* terbaru untuk 'membawa' komponen NFV ke layanan *cloud*[14]. Metode ini dinilai memiliki keunggulan dibanding menggunakan *virtual machine* atau VM yang mana kontainerisasi tidak mengontrol sistem secara penuh layaknya VM, sehingga membuat mereka menjadi lebih ringan dan berefek pada waktu *deployment* ke layanan *cloud* yang lebih cepat. Aplikasi kontainer yang digunakan di

penelitian ini yaitu *Docker* dengan bantuan *plug in* yang dimiliki oleh *Docker* sendiri yaitu *Docker Compose* sebagai fitur yang dapat menjalankan banyak kontainer dalam satu waktu dan ini diperlukan mengingat banyaknya komponen dalam arsitektur jaringan 5G terutama di bagian *core*.

D. Open-Source 5G Network Software

Dalam penelitian ini salah 5G *core* yang digunakan adalah Free5GC merupakan sebuah proyek *open source* untuk mengembangkan *core network* jaringan 5G. Proyek ini memiliki tujuan untuk dapat mengimplementasikan 5G *core network* atau 5GC sesuai standar 3GPP Release 15[15].



Gambar 2.3 Logo Free5GC

Selain menggunakan Free5GC sebagai *core network*, pada penelitian ini penulis menggunakan 5G *core* lain yaitu OpenAirInterface atau biasa disingkat dengan OAI yang memiliki fungsi sama dengan Free5GC untuk menjalankan simulasi dan dilakukan analisis QoS untuk dibandingkan satu sama lain.[16]



Gambar 2.4 Logo Open Air Interface

UERANSIM merupakan sebuah *open source* proyek untuk pengembangan 5G sebagai simulator di sisi pengguna. UERANSIM bekerja sebagai *User Equipment* (UE) 5G atau pelanggan yang diibaratkan simulasi menggunakan telepon genggam 5G dan juga sebagai *Radio Access Network* (RAN) atau diimplementasikan sebagai *base station* yang bisa memancarkan jaringan 5G ke pelanggan[15]. UERANSIM mensimulasikan bagian *access network* 5G khususnya untuk jaringan Free5GC.



Gambar 2.5 Logo UERANSIM

E. Quality of Services (QoS)

1. Delay

Delay merupakan perhitungan waktu yang dibutuhkan sebuah data untuk mencapai jarak dari asal ke tujuan. *Delay* biasanya menjadi acuan sebuah kualitas layanan karena banyak faktor yang memengaruhi parameter ini yang diantaranya dipengaruhi oleh media fisik, waktu proses, dan jarak[6]. Untuk tingkat kualitas *delay* menurut TIPHON ETSI (1999-2006) suatu jaringan seperti pada Tabel 2.1 [17] berikut:

Kategori Delay	Indeks	Delay (ms)
Sangat Bagus	4	< 150
Bagus	3	150 - 300
Cukup Bagus	2	300 - 450
Buruk	1	> 450

TABEL 2.1 Standar Kualitas Delay Menurut ETSI

$$\text{Delay (Rata - Rata)} = \frac{\text{Total Delay}}{\text{Total Jumlah Paket Diterima}}$$

2. Jitter

Jitter adalah variasi dari *delay*, hal yang memengaruhi *jitter* yaitu beban dari trafik dalam jaringan tersebut, kemudian faktor lainnya adalah ukuran tumpukan paket yang terjadi dalam satu jaringan[6]. Kualitas *jitter* suatu jaringan menurut TIPHON ETSI (1999 – 2006) bisa kita lihat di Tabel 2.2 [17] seperti berikut:

Kategori Jitter	Indeks	Jitter (ms)
Sangat Bagus	4	0
Bagus	3	0 - 75
Cukup Bagus	2	75 - 125
Buruk	1	125 - 225

TABEL 2.2 Standar Kualitas Jitter Menurut ETSI

$$\text{Total Variasi Delay} = \text{Delay} - (\text{Rata-Rata Delay})$$

$$\text{Jitter} = \frac{\text{Total Variasi Delay}}{\text{Total Jumlah Paket yang Diterima}}$$

3. Packet Loss

Packet Loss adalah parameter yang menghitung keadaan di mana suatu paket hilang. Beberapa faktor menjadi penyebab terjadinya suatu paket hilang, diantaranya yaitu karena terjadi eror di media fisik jaringan, kemudian terjadi tabrakan (*overload*) pada *traffic* jaringan tersebut[6]. Dalam *packet loss*, semakin kecil persentase yang didapat maka semakin baik kualitasnya. Kualitas *packet loss* suatu jaringan menurut TIPHON ETSI bisa dilihat pada Tabel 2.3 [17] berikut:

TABEL 2.3 Standar Kualitas Packet Loss Menurut ETSI

Kategori Packet Loss	Indeks	Persentase Packet Loss (%)
Sangat Bagus	4	0 – 3
Bagus	3	3 – 15
Cukup Bagus	2	15 – 25
Buruk	1	> 25

$$\text{Packet Loss} = \frac{(\text{Paket Dikirim} - \text{Paket Diterima})}{\text{Paket Diterima}} \times 100\%$$

4. Throughput

Throughput adalah parameter yang menghitung kecepatan data efektif dalam suatu jaringan, kecepatan data yang dihitung tersebut memiliki satuan yaitu *bit per second* (bps)[6]. *Throughput* biasanya menjadi kebalikan dari *packet loss*, ia menghitung jumlah paket yang

berhasil terkirim ke tujuan. Kualitas performa *throughput* menurut TIPHON ETSI bisa dilihat di Tabel 2.4[17] berikut:

TABEL 2.4 Standar Kualitas Throughput Menurut ETSI

Kategori Throughput	Indeks	Throughput
Sangat Bagus	4	>2,1 Mbps
Bagus	3	1200 kbps – 2,1 Mbps
Cukup Bagus	2	700 kbps – 1200 kbps
Buruk	1	0 kbps -700 kbps

$$\text{Throughput} = \frac{\text{Paket Data Diterima}}{\text{Waktu Pengamatan}}$$

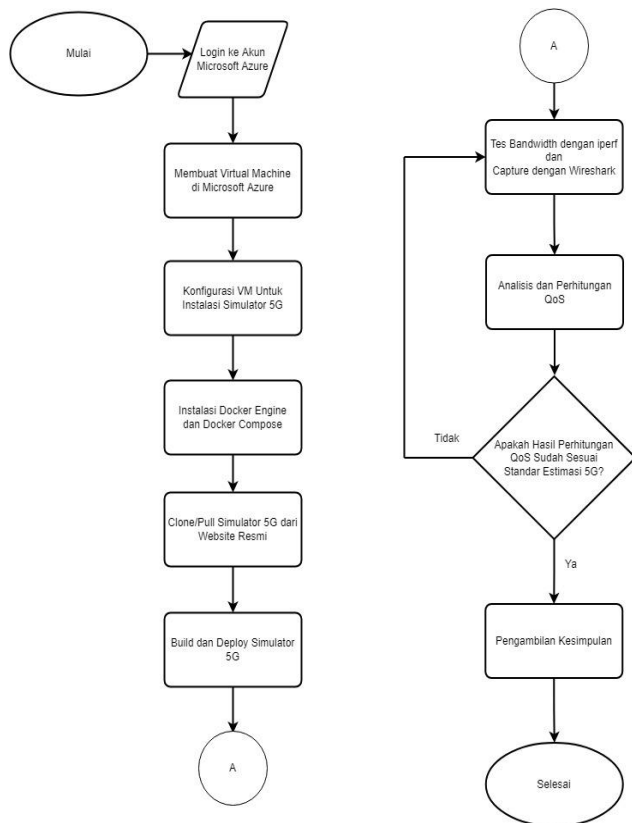
Keempat parameter di atas merupakan acuan parameter QoS suatu jaringan untuk berbagai layanan. Untuk melakukan pengukuran QoS pada jaringan 5G ini, penulis menggunakan bantuan beberapa *software* khusus untuk pengukuran jaringan yaitu Wireshark dan IPerf. Wireshark merupakan sebuah perangkat lunak yang berfungsi untuk menangkap dan menganalisis suatu paket data di sebuah jaringan. Dengan wireshark kita bisa menangkap sebuah paket yang sedang dikirimkan lalu dianalisis isi dari paket tersebut[19], selain itu keuntungan menggunakan wireshark sendiri yaitu karena ia bersifat terbuka atau *open source* untuk lisensinya.

Sementara IPerf merupakan sebuah perangkat lunak yang digunakan untuk mengukur performansi suatu jaringan dengan melakukan *tracking* dari setiap paket yang dikirim dalam jaringan tersebut[20]. Cara kerja IPerf ini yaitu dengan memasukkan perintah menjadi *client* dan server pada setiap mesin yang akan diuji, dalam penelitian ini *core network* bersifat sebagai server dan *access network* sebagai *client*. Pengukuran sendiri dilakukan selama 15 detik dengan skema mengirim paket data dan *capturing* paket yang dikirimkan melalui protokol TCP.

III. METODE

A. Diagram Alir Penelitian

Penelitian ini dilakukan secara berkelompok dengan membagi fokus pada implementasi jaringan 5G dan melakukan analisis pada sistem dengan melihat dari aspek kualitas layanan dan juga aspek keamanan yang dilakukan analisis oleh rekan sekelompok penulis. Setelah jaringan berhasil terbentuk dan dijalankan, penulis baru bisa melakukan analisis serta perbandingan tata cara instalasi melalui *virtual machine* biasa atau kontainer. Penulis perlu melakukan banyak konfigurasi untuk membangun jaringan ini mengingat masih sangat minimnya penelitian serupa yang dilakukan. Dalam penelitian ini penulis telah berhasil mengimplementasikan dan menjalankan jaringan dengan alur pengerjaan serta analisis seperti terangkum dan tergambar pada Gambar 3.1 di bawah ini.

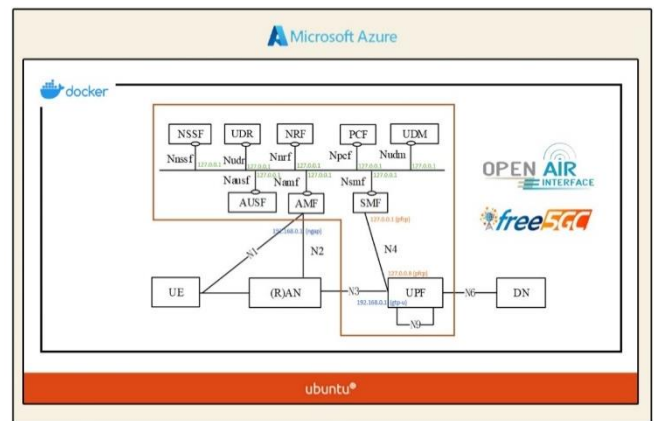


GAMBAR 3.1 Diagram Alir Penelitian Jaringan 5G Privat

B. Desain Sistem

Dalam penelitian ini, dilakukan sebuah simulasi jaringan 5G yang berjenis *standalone* yang telah diuji melalui dua skenario penempatan komponen dan penggunaannya. Dalam arsitektur jaringan 5G kita bisa melihat ada beberapa komponen yang masuk dalam ranah *Control Plane (C-Plane)* dan *User Plane (U-Plane)*, kedua ranah tersebut memisahkan komponen dalam jaringan *core* 5G yang berasal dari *software* simulasi Free5GC. Komponen yang masuk dalam ranah *Control Plane* sendiri yaitu semua komponen yang ada di *core network* mulai dari NSSF, NEF, NRF, PCF, UDM, AF, AUSF, hingga AMF yang setiap komponen memiliki fungsinya masing-masing seperti yang telah dijelaskan pada bab sebelumnya.

Nantinya setiap komponen *Control Plane* ini terhubung dengan UPF yang terkoneksi dengan RAN dan juga UE di ranah *access network* dari jaringan ini dan masuk dalam ranah *User Plane* jaringan ini. Setelah semua berhasil terintegrasi dengan baik maka kita bisa melakukan koneksi dari simulator UE pada *access network*. Semua komponen yang ada pada sisi *core* maupun *access network* tersimpan dalam dua buah *virtual machine* yang masing-masingnya diisi oleh salah satu bagian jaringan tersebut. Kedua *virtual machine* tersebut disimpan secara lokal pada platform *VirtualBox* dan secara komputasi awan disimpan dan disebar melalui platform layanan *cloud* yaitu Microsoft Azure seperti yang divisualisasikan pada Gambar 3.2.



GAMBAR 3.2 Model Sistem Jaringan 5G Privat di Layanan Cloud Computing

Semua skenario penempatan tersebut selanjutnya dilakukan pengukuran *Quality of Services (QoS)* yang meliputi parameter mulai dari *throughput*, *delay*, *jitter*, hingga *packet loss*. Perhitungan QoS ini akan menjadi tolak ukur bagi penulis untuk mendapatkan hasil performansi jaringan 5G privat pada skenario pada komputer lokal maupun skenario pada layanan *cloud computing* tepatnya pada platform Microsoft Azure.

Kedua model di atas menggambarkan kinerja sistem pada kedua skenario, terlihat perbedaan pada skenario *cloud computing* terdapat penggabungan jalur jaringan ke satu *virtual network* yang mana hal ini sendiri merupakan bentuk skema atau konsep yang ada pada layanan Microsoft Azure. Perbedaan jalur jaringan tersebut menjadi pengaruh bagi paket data yang dikirim karena skema yang berbeda dengan skema jalur jaringan yang ada pada lokal komputer.

C. Perangkat Keras Pendukung

Berikut ini merupakan perangkat keras pendukung sistem yang menopang dua skenario pada layanan *cloud computing*, spesifikasi perangkat di bawah merupakan spesifikasi server milik Microsoft yang digunakan untuk Microsoft Azure. Berikut spesifikasi server D-Series milik Microsoft.

Prosesor	Intel XEON E5-2673 v4 (2 Vcpu) – 2.3 GHz
RAM	2.00 GB
Penyimpanan	50 GB
Lokasi Server	Korea Central (Azure Server Asia Pasifik)

TABEL 3.1 Spesifikasi Komputer Server D-Series Microsoft

IV. HASIL DAN PEMBAHASAN

A. Hasil Pengukuran Kualitas Layanan Free5GC

Berikut ini merupakan hasil pengujian *Quality of Services* pada jaringan Free5GC di layanan *cloud* Microsoft Azure. Dari pengujian 1 UE, 3 UE, dan 5 UE yang diuji dengan mengirimkan paket data melalui tes *bandwidth* dari IPerf, didapatkan perhitungan QoS dengan detail parameter sebagai berikut ini.

➤ Pengujian 1 UE

• Throughput

$$\begin{aligned}
 \text{Throughput} &= \frac{\text{Paket Data Diterima}}{\text{Waktu Pengamatan}} \\
 &= \frac{3711088084 \text{ byte}}{15.008 \text{ second}} \\
 &= 247273992.804 \text{ byte/s (Bps)} \\
 &= 1978191942 \text{ bit/s (bps)} \\
 &= \mathbf{1.9782 \text{ Gigabit/s (Gbps)}}
 \end{aligned}$$

• Packet Loss

$$\begin{aligned}
 \text{Packet Loss} &= \frac{(\text{Pkt. Dikirim} - \text{Pkt. Diterima})}{\text{Pkt. Diterima}} \times 100\% \\
 &= \frac{(134652 - 134652)}{134652} \times 100\% \\
 &= \mathbf{0\%}
 \end{aligned}$$

• Delay

$$\begin{aligned}
 \text{Delay} &= \frac{\text{Total Delay}}{\text{Total Jumlah Paket Diterima}} \\
 &= \frac{15.008}{134652} \\
 &= 0.000114576835 \text{ second} \\
 &= \mathbf{0.11145768351 \text{ milisecond (ms)}}
 \end{aligned}$$

• Jitter

$$\begin{aligned}
 \text{Total Variasi Delay} &= \text{Delay} - (\text{Rata} - \text{Rata Delay}) \\
 &= 15.008 - 0.000114576835 \\
 &= \mathbf{15.0078885423165 \text{ second}}
 \end{aligned}$$

$$\begin{aligned}
 \text{Jitter} &= \frac{\text{Total Variasi Delay}}{\text{Total Jumlah Paket yang Diterima}} \\
 &= \frac{15.0078885423165}{134652} \\
 &= \mathbf{0.00011145686 \text{ second}} \\
 &= \mathbf{0.111456856 \text{ milisecond (ms)}}
 \end{aligned}$$

➤ Pengujian 3 UE

• Throughput

$$\begin{aligned}
 \text{Throughput} &= \frac{\text{Paket Data Diterima}}{\text{Waktu Pengamatan}} \\
 &= \frac{4220580418 \text{ byte}}{18.036 \text{ second}} \\
 &= 234008672.544 \text{ byte/s (Bps)} \\
 &= 1872069380 \text{ bit/s (bps)} \\
 &= \mathbf{1.8721 \text{ Gigabit/s (Gbps)}}
 \end{aligned}$$

• Packet Loss

$$\begin{aligned}
 \text{Packet Loss} &= \frac{(\text{Paket Dikirim} - \text{Paket Diterima})}{\text{Paket Diterima}} \times 100\% \\
 &= \frac{(120955 - 120955)}{120955} \times 100\% \\
 &= \mathbf{0\%}
 \end{aligned}$$

• Delay

$$\begin{aligned}
 \text{Delay} &= \frac{\text{Total Delay}}{\text{Total Jumlah Paket Diterima}} \\
 &= \frac{18.036}{120955} \\
 &= 0.0001491133066 \text{ second} \\
 &= \mathbf{0.14911330660 \text{ milisecond (ms)}}
 \end{aligned}$$

• Jitter

$$\begin{aligned}
 \text{Total Variasi Delay} &= \text{Delay} - (\text{Rata} - \text{Rata Delay}) \\
 &= 18.036 - 0.0001491133066 \\
 &= \mathbf{18.0358508866934 \text{ second}}
 \end{aligned}$$

$$\begin{aligned}
 \text{Jitter} &= \frac{\text{Total Variasi Delay}}{\text{Total Jumlah Paket yang Diterima}} \\
 &= \frac{18.0358508866934}{120955} \\
 &= \mathbf{0.00014911207 \text{ second}} \\
 &= \mathbf{0.149112074 \text{ milisecond (ms)}}
 \end{aligned}$$

➤ Pengujian 5 UE

• Throughput

$$\begin{aligned}
 \text{Throughput} &= \frac{\text{Paket Data Diterima}}{\text{Waktu Pengamatan}} \\
 &= \frac{5063428100 \text{ byte}}{24.416 \text{ second}} \\
 &= 207381557.176 \text{ byte/s (Bps)} \\
 &= 1659052457 \text{ bit/s (bps)} \\
 &= \mathbf{1.6591 \text{ Gigabit/s (Gbps)}}
 \end{aligned}$$

• Packet Loss

$$\begin{aligned}
 \text{Packet Loss} &= \frac{(\text{Paket Dikirim} - \text{Paket Diterima})}{\text{Paket Diterima}} \times 100\% \\
 &= \frac{(138388 - 138388)}{138388} \times 100\% \\
 &= \mathbf{0\%}
 \end{aligned}$$

• Delay

$$\begin{aligned}
 \text{Delay} &= \frac{\text{Total Delay}}{\text{Total Jumlah Paket Diterima}} \\
 &= \frac{24.416}{138388} \\
 &= 0.0001764314825 \text{ second} \\
 &= \mathbf{0.17643148250 \text{ milisecond (ms)}}
 \end{aligned}$$

• Jitter

$$\begin{aligned}
 \text{Total Variasi Delay} &= \text{Delay} - (\text{Rata} - \text{Rata Delay}) \\
 &= 24.416 - 0.0001764314825 \\
 &= \mathbf{24.4158235685175 \text{ second}}
 \end{aligned}$$

$$\text{Jitter} = \frac{\text{Total Variasi Delay}}{\text{Total Jumlah Paket yang Diterima}}$$

$$= \frac{24.4158235685175}{138388}$$

$$= \mathbf{0.00017643021 \text{ second}}$$

$$= \mathbf{0.176430208 \text{ milisecond (ms)}}$$

B. Hasil Pengukuran Kualitas Layanan OpenAirInterface

Berikut ini merupakan hasil pengujian *Quality of Services* pada jaringan OpenAirInterface di layanan *cloud* Microsoft Azure. Dari pengujian 1 UE, 3 UE, dan 5 UE yang diuji dengan mengirimkan paket data melalui tes *bandwidth* dari IPerf, didapatkan perhitungan QoS dengan detail parameter sebagai berikut ini.

➤ Pengujian 1 UE

• Throughput

$$\begin{aligned} \text{Throughput} &= \frac{\text{Paket Data Diterima}}{\text{Waktu Pengamatan}} \\ &= \frac{2923164482 \text{ byte}}{15.023 \text{ second}} \\ &= 194579277.242 \text{ byte/s (Bps)} \\ &= 1556634218 \text{ bit/s (bps)} \\ &= \mathbf{1.5566 \text{ Gigabit/s (Gbps)}} \end{aligned}$$

• Packet Loss

$$\begin{aligned} \text{Packet Loss} &= \frac{(\text{Pkt. Dikirim} - \text{Pkt. Diterima})}{\text{Pkt. Diterima}} \times 100\% \\ &= \frac{(97679 - 97679)}{97679} \times 100\% \\ &= \mathbf{0\%} \end{aligned}$$

• Delay

$$\begin{aligned} \text{Delay} &= \frac{\text{Total Delay}}{\text{Total Jumlah Paket Diterima}} \\ &= \frac{15.023}{97679} \\ &= 0.0001537996908 \text{ second} \\ &= \mathbf{0.15379969082 \text{ milisecond (ms)}}$$

• Jitter

$$\begin{aligned} \text{Total Variasi Delay} &= \text{Delay} - (\text{Rata} - \text{Rata Delay}) \\ &= 15.023 - 0.0001537996908 \\ &= \mathbf{15.0228462003092 \text{ second}} \end{aligned}$$

$$\begin{aligned} \text{Jitter} &= \frac{\text{Total Variasi Delay}}{\text{Total Jumlah Paket yang Diterima}} \\ &= \frac{15.0228462003092}{97679} \\ &= \mathbf{0.00015379812 \text{ second}} \\ &= \mathbf{0.153798116 \text{ milisecond (ms)}}$$

➤ Pengujian 3 UE

• Throughput

$$\begin{aligned} \text{Throughput} &= \frac{\text{Paket Data Diterima}}{\text{Waktu Pengamatan}} \\ &= \frac{2833427348 \text{ byte}}{15.009 \text{ second}} \end{aligned}$$

$$= 188781887.401 \text{ byte/s (Bps)}$$

$$= 1510255099 \text{ bit/s (bps)}$$

$$= \mathbf{1.5103 \text{ Gigabit/s (Gbps)}}$$

• Packet Loss

$$\begin{aligned} \text{Packet Loss} &= \frac{(\text{Paket Dikirim} - \text{Paket Diterima})}{\text{Paket Diterima}} \times 100\% \\ &= \frac{(74960 - 74960)}{74960} \times 100\% \\ &= \mathbf{0\%} \end{aligned}$$

• Delay

$$\begin{aligned} \text{Delay} &= \frac{\text{Total Delay}}{\text{Total Jumlah Paket Diterima}} \\ &= \frac{15.009}{74960} \\ &= 0.0002002267876 \text{ second} \\ &= \mathbf{0.20022678762 \text{ milisecond (ms)}}$$

• Jitter

$$\begin{aligned} \text{Total Variasi Delay} &= \text{Delay} - (\text{Rata} - \text{Rata Delay}) \\ &= 15.009 - 0.0002002267876 \\ &= \mathbf{15.0087997732124 \text{ second}} \end{aligned}$$

$$\begin{aligned} \text{Jitter} &= \frac{\text{Total Variasi Delay}}{\text{Total Jumlah Paket yang Diterima}} \\ &= \frac{15.0087997732124}{74960} \\ &= \mathbf{0.00020022412 \text{ second}} \\ &= \mathbf{0.200224117 \text{ milisecond (ms)}}$$

➤ Pengujian 5 UE

• Throughput

$$\begin{aligned} \text{Throughput} &= \frac{\text{Paket Data Diterima}}{\text{Waktu Pengamatan}} \\ &= \frac{2798844976 \text{ byte}}{15.676 \text{ second}} \\ &= 178543313.090 \text{ byte/s (Bps)} \\ &= 1428346505 \text{ bit/s (bps)} \\ &= \mathbf{1.4283 \text{ Gigabit/s (Gbps)}}$$

• Packet Loss

$$\begin{aligned} \text{Packet Loss} &= \frac{(\text{Paket Dikirim} - \text{Paket Diterima})}{\text{Paket Diterima}} \times 100\% \\ &= \frac{(72144 - 72144)}{72144} \times 100\% \\ &= \mathbf{0\%} \end{aligned}$$

• Delay

$$\begin{aligned} \text{Delay} &= \frac{\text{Total Delay}}{\text{Total Jumlah Paket Diterima}} \\ &= \frac{15.676}{72144} \\ &= 0.0002172876469 \text{ second} \end{aligned}$$

$$= 0.21728764693 \text{ milisecond (ms)}$$

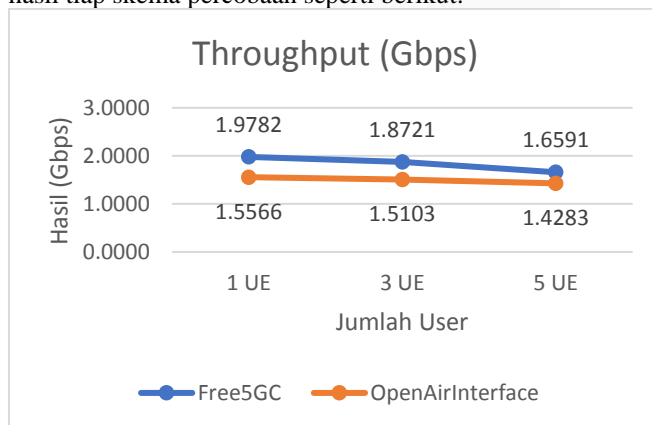
- Jitter
Total Variasi Delay = Delay – (Rata – Rata Delay)
= 15.676 – 0.0002172876469
= **15.6757827123531 second**

$$\begin{aligned} \text{Jitter} &= \frac{\text{Total Variasi Delay}}{\text{Total Jumlah Paket yang Diterima}} \\ &= \frac{15.6757827123531}{72144} \\ &= \mathbf{0.00021728464 \text{ second}} \\ &= \mathbf{0.217284635 \text{ milisecond (ms)}} \end{aligned}$$

C. Analisis Perbandingan Kualitas Layanan

Dari hasil perhitungan kualitas layanan atau QoS di atas telah didapatkan nilai dari setiap parameter yang meliputi Throughput, Packet Loss, Delay, dan Jitter. Setiap hasil tersebut telah dibandingkan satu sama lain dan dimasukkan ke parameter tabel kualitas berdasarkan standarisasi yang dilakukan oleh TIPHON ETSI. Hasil perhitungan tersebut juga digunakan untuk simulasi standarisasi QoS pada layanan multimedia. Hasil perhitungan ini dilakukan dalam jaringan 5G yang berarti bukan pada jaringan aslinya, namun perhitungan ini bisa digunakan sebagai tolak ukur dalam melihat performansi dari jaringan tersebut.

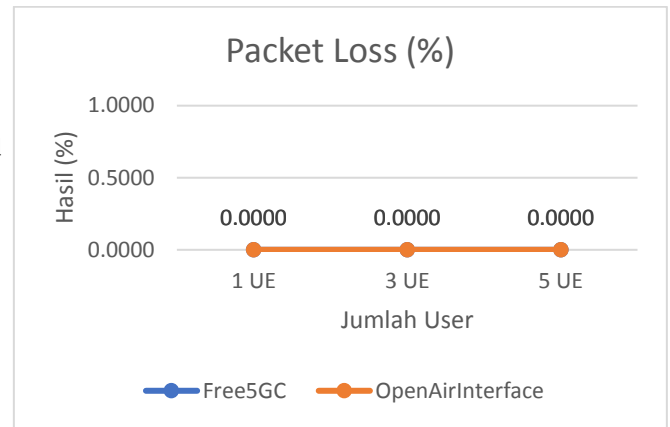
Berikut ini perbandingan perhitungan QoS pada jaringan 5G OpenAirInterface dengan Free5GC. Perbandingan dilakukan pada setiap parameter yang meliputi hasil tiap skema percobaan seperti berikut:



GRAFIK 4.1 Perbandingan Throughput Kedua Jaringan

Perhitungan Throughput di kedua jaringan berjalan dengan lancar saat dilakukan tes. Pada grafik 4.1 di atas terlihat bahwa hasil perhitungan Throughput OpenAirInterface terlihat lebih stabil saat dilakukan penambahan UE atau pengguna. Sementara itu pada Free5GC hasil yang didapat menurun saat dilakukan penambahan UE. Dapat dilihat saat penambahan dari 1 UE ke 3 UE adanya penurunan yang cukup signifikan saat penambahan dari 1 sampai 5 UE. Pengiriman paket ini menjadi pengaruh bagi jaringan dalam memberikan performansi terbaiknya.

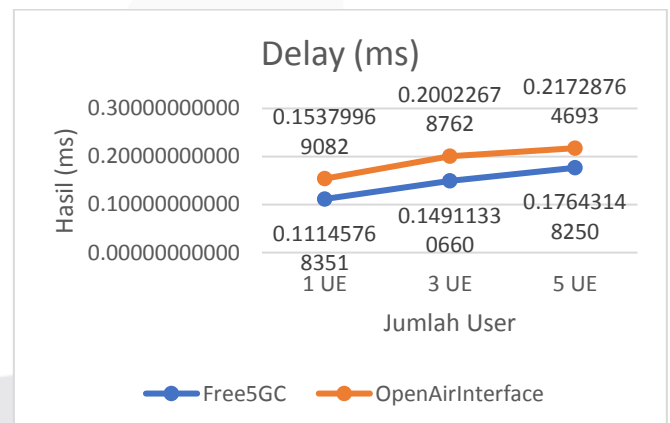
Selanjutnya parameter yang dilakukan perhitungan adalah Packet Loss. Parameter ini sangat berfungsi menggambarkan jumlah paket yang gagal dikirim ke alamat tujuan, berikut detail hasil pengukuran packet loss:



GRAFIK 4.2 Perbandingan Packet Loss Kedua Jaringan

Kedua jaringan mendapatkan hasil yang sama dalam pengukuran QoS parameter Packet Loss. Seperti kita lihat pada grafik 4.2 didapat hasil sebesar 0% bagi semua skema baik pada Free5GC maupun OpenAirInterface. Hal tersebut dikarenakan karena tes ini dilakukan pada jaringan yang kemungkinannya kecil dalam terjadi packet loss yang disebabkan faktor-faktor lain. Kemudian waktu tes yang hanya 15 detik memungkinkan tidak terjadinya kesalahan karena tergolong sebentar saja.

Selanjutnya parameter Delay, dalam jaringan ini perbandingan Delay dapat dilihat dalam grafik 4.3 berikut ini:

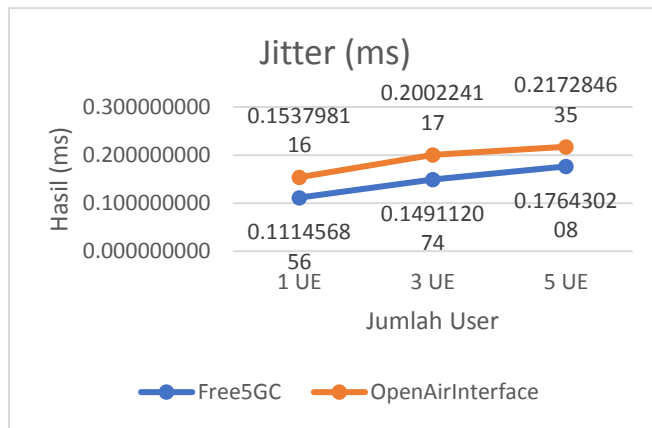


GRAFIK 4.3 Perbandingan Delay Kedua Jaringan

Mengacu pada grafik 4.3 di atas, perhitungan Delay pada kedua jaringan berjalan dengan lancar. Hasil yang didapat juga cenderung memiliki perbedaan, untuk jaringan OpenAirInterface dapat dilihat bahwa ada kenaikan yang signifikan ketika terjadi penambahan pengguna dari 1 sampai 5 UE. Sementara pada jaringan Free5GC hasil yang didapat cenderung stabil dan tidak ada kenaikan atau penurunan yang signifikan saat dilakukan penambahan pengguna dari 1 sampai 5 UE. Jumlah paket yang dikirimkan menjadi pengaruh dalam waktu atau Delay setiap paket, hal tersebut bisa dilihat dalam total Delay yang didapatkan dan memiliki jarak cukup jauh untuk setiap skema.

Selanjutnya parameter terakhir yang dilakukan pengukuran adalah Jitter. Parameter ini menggambarkan variasi dari setiap waktu pengiriman paket yang pastinya memiliki selisih dengan waktu pengiriman paket lain. Semua variasi Delay tersebut dirata-ratakan untuk setiap paket.

Berikut ini perbandingan hasil pengukuran Jitter di kedua jaringan yang tertera pada grafik 4.4 berikut ini:



GRAFIK 4.4 Perbandingan Jitter Kedua Jaringan

Hasil perbandingan dari perhitungan Jitter pada grafik 4.4 di atas menunjukkan kemiripan dengan bentuk grafik 4.3 untuk perhitungan Delay. Hal tersebut sebuah kewajaran karena Jitter merupakan variasi Delay yang pastinya perhitungannya tidak jauh berbeda terutama rata-rata yang didapat. Sama seperti Delay, perhitungan pada jaringan OpenAirInterface terlihat ada kenaikan yang cukup signifikan ketika ada penambahan dari 1 sampai 5 UE. Sementara pada jaringan Free5GC terlihat grafik yang cenderung stabil sekalipun dilakukan penambahan pengguna dari 1 sampai 5 UE. Pada jaringan ini terlihat jarak pengiriman setiap paket tidak terlalu signifikan dan hal tersebut memengaruhi terhadap variasi delay atau Jitter ini.

V. KESIMPULAN

Dari hasil penelitian di atas didapatkan beberapa kesimpulan mengenai analisis jaringan 5G privat di layanan *cloud computing*. Berikut kesimpulan yang didapatkan penulis dalam beberapa poin berikut:

- A. Berhasil membangun atau mengimplementasikan *core network* 5G dengan jaringan Free5GC dan OpenAirInterface di layanan *cloud computing* Microsoft Azure dengan bantuan *deployment* metode containerisasi.
- B. Alur pembuatan sistem yang dibantu menggunakan *build image* kontainer di Docker memengaruhi proses membangun dan menjalankan sistem. Konsep ini digunakan pada kedua jaringan baik Free5GC maupun OpenAirInterface. Hal tersebut dikarenakan *plugin* Docker Compose dapat menjalankan banyak kontainer komponen 5G dalam satu waktu.
- C. Hasil pengukuran *Quality of Services* pada jaringan OpenAirInterface dengan skenario 1,3, dan 5 UE didapat Throughput sebesar 1.5566 Gbps, 1.5103 Gbps, dan 1.4283 Gbps. Untuk Delay didapat 0.15379969082 ms, 0.20022678762 ms, dan 0.21728764693 ms. Lalu Jitter didapat 0.153798116 ms, 0.200224117 ms, dan 0.217284635 ms. Terakhir, untuk Packet Loss didapat hasil 0% atau tidak ada.
- D. Hasil pengukuran *Quality of Services* pada jaringan Free5GC dengan skenario 1,3, dan 5 UE didapat Throughput sebesar 1.9782 Gbps, 1.8721 Gbps, dan 1.6591 Gbps. Untuk Delay didapat 0.11145768351 ms,

0.14911330660 ms, dan 0.17643148250 ms. Lalu Jitter didapat 0.111456856 ms, 0.149112074 ms, dan 0.176430208 ms. Terakhir, untuk Packet Loss didapat hasil 0% atau tidak ada.

- E. Dalam pengukuran ini, kedua jaringan baik Free5GC maupun OpenAirInterface memiliki performansi yang baik. Namun dari hasil yang didapat, dapat disimpulkan Free5GC terbilang lebih stabil dan lebih baik dibanding OpenAirInterface.
- F. Layanan *cloud* Microsoft Azure disimpulkan belum bisa digunakan untuk infrastruktur jaringan khususnya secara privat. Penyebabnya yaitu ada beberapa kebijakan dan aturan dari Microsoft Azure yang tidak mendukung penulis untuk mengembangkan secara maksimal. Namun dari sisi *cost*, Azure terhitung lebih murah dibanding layanan lainnya.

REFERENSI

- [1] E. Goshi, M. Jarschel, R. Pries, M. He, and W. Kellerer, "Investigating Inter-NF Dependencies in Cloud-Native 5G Core Networks."
- [2] F. Kaltenberger, A. P. Silva, A. Gosain, L. Wang, and T.-T. Nguyen, "OpenAirInterface: Democratizing Innovation in the 5G Era," 2020.
- [3] G. Mao, "5G Green Mobile Communication Networks BOOK REVIEW," 2017.
- [4] O. W. Purbo, K. Muludi, and T. C. Kurniawan, *Jaringan Nirkabel 5G Berbasis Cloud - Reability, Mobility, Energy Efficiency, Latency*. Yogyakarta: ANDI, 2020.
- [5] S. Liu, Z. Xu, and Z. Tian, "Implementation of NRF in the Docker-based NFV platform," *The Journal of Engineering*, vol. 2019, no. 23, pp. 8884–8887, Dec. 2019, doi: 10.1049/joe.2018.9135.
- [6] L. K. S, A. R. Shankar, P. Student, R. scholar, and A. Professor, "QoS Analysis for 5G Networks," 2018.
- [7] J. Krzywdka, W. Tärneberg, P. Östberg, M. Kihl, and E. Elmroth, "Telco Clouds: Modelling and Simulation," 2015. [Online]. Available: <http://www.diva-portal.org/http://urn.kb.se/resolve?urn=urn:nbn:se:umu:diva-104688>
- [8] H. T. S. AlRikabi, A. H. M. Alaidi, A. S. Abdalrada, and F. T. Abed, "Analysis of the efficient energy prediction for 5G wireless communication technologies," *International Journal of Emerging Technologies in Learning*, vol. 14, no. 8, pp. 23–37, 2019, doi: 10.3991/ijet.v14i08.10485.
- [9] M. Wen *et al.*, "Private 5G Networks: Concepts, Architectures, and Research Landscape," *IEEE Journal on Selected Topics in Signal Processing*, vol.

- 16, no. 1, pp. 7–25, Jan. 2022, doi: 10.1109/JSTSP.2021.3137669.
- [10] TSGS, “TS 123 501 - V15.2.0 - 5G; System Architecture for the 5G System (3GPP TS 23.501 version 15.2.0 Release 15),” 2018. [Online]. Available: <https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx>
- [11] Carlos Alberto Kamienski and Alexandre Heideker, *Managing Elasticity in an NFV-based IaaS Environment*. 2017.
- [12] S. Arora and A. Ksentini, “Dynamic Resource Allocation and Placement of Cloud Native Network Services,” 2021.
- [13] V. B. Gandhi H Gardi, C. K. Kumbharana, V. A. Gandhi, and C. K. Kumbharana Head, “Comparative study of Amazon EC2 and Microsoft Azure cloud architecture International Journal of Advanced Networking Applications (IJANA) Comparative study of Amazon EC2 and Microsoft Azure cloud architecture,” 2018, [Online]. Available: <https://www.researchgate.net/publication/327537294>
- [14] C. Pahl, A. Brogi, J. Soldani, and P. Jamshidi, “Cloud container technologies: A state-of-the-art review,” *IEEE Transactions on Cloud Computing*, vol. 7, no. 3, pp. 677–692, Jul. 2019, doi: 10.1109/TCC.2017.2702586.
- [15] Z. Min, S. Shekhar, C. Mahmoudi, V. Formicola, S. Gokhale, and A. Gokhale, “Dynamic 5G Network Slice Management Middleware for Industrial Internet of Things: Industry Paper,” Jul. 2022, [Online]. Available: <http://arxiv.org/abs/2207.07219>
- [16] F. Kaltenberger, G. de Souza, R. Knopp, and H. Wang, “The OpenAirInterface 5G New Radio Implementation: Current Status and Roadmap.” [Online]. Available: <https://gitlab.eurecom.fr/oai/openairinterface5g/wiki/s/>
- [17] E. European Telecommunication Standard Institute, “Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON); General aspects of Quality of Service (QoS),” 1999. [Online]. Available: <http://www.etsi.org>
- [18] A. Zainuddin and L. A. Syamsul Irfan, “Analisis QoS (Quality of Service) Jaringan Internet Kampus (Studi Kasus : Fakultas Teknik Universitas Matram).”
- [19] M. Risnandar *et al.*, “IMPLEMENTASI VOICE OVER INTERNET PROTOCOL (VOIP) BERBASIS SESSION INITIATION PROTOCOL (SIP) BERBANTUAN BRIKER VERSI 1.4 UNTUK PENGUKURAN QUALITY OF SERVICES PADA JARINGAN KOMPUTER DI FAKULTAS TEKNIK UIKA BOGOR.”
- [20] I. Putu, A. E. Pratama, I. Made, and A. Wikantyasa, “Implementasi dan Analisis Simulasi QOS dan Performace Device dengan Menggunakan ONOS dan Iperf3,” vol. 4, no. 2, pp. 2622–4615, 2019, [Online]. Available: <http://openjournal.unpam.ac.id/index.php/informatika57>